

Также с помощью системы составляются результирующие таблицы (по каждому серверу трехуровневой архитектуры), графики и диаграммы по данным анализаторов трафика.

ID	IP	tcp_port	delta_time	time	url
1	192.168.122.9	51017	0.038872	0:38:17.119518	http://avtokomplekt66.ru/
2	192.168.122.9	51019	0.098872	0:38:17.528404	http://avtokomplekt66.ru/traktora

Рис. 2. Результирующая таблица

Общая схема функционирования системы нагрузочного тестирования представлена на рис. 3.

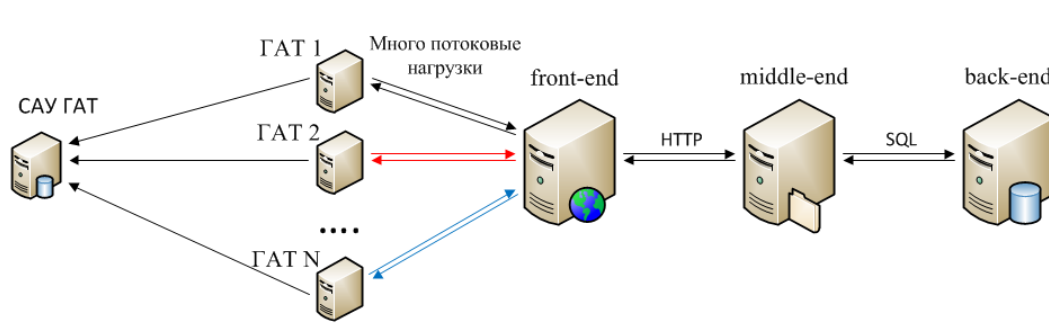


Рис. 3. Схема функционирования системы нагрузочного тестирования

«ГАТ N» – генератор и анализатор трафика. Ключевой элемент системы, позволяющий генерировать много потоковый трафик, а также анализировать входящий трафик и отправлять результат на CAU GAT.

В итоге используя данную систему, возможно, обеспечить любую требуемую нагрузку, изменяя лишь количество ГАТ. По разнице времени запроса генератора и ответа веб-сайта, а также URL адресу, можно судить справляется ли сервер с данной нагрузкой, или нет. Стоит отметить, что разница во времени будет иметь незначительную погрешность на время пока сигнал идет от генератора до сервера и обратно, которую можно приравнять к нулю (погрешность еще уменьшится, если вместо URL адреса сайта, прописывать его IP адрес при генерации).

#### Список использованных источников

1. Википедия, свободная энциклопедия [Электронный ресурс]. Нагрузочное тестирование. Режим доступа: <http://ru.wikipedia.org/wiki/>.
2. Википедия, свободная энциклопедия [Электронный ресурс]. Трехуровневая архитектура. Режим доступа: <http://ru.wikipedia.org/wiki/>.

### АВТОМАТИЗАЦИЯ ТЕЛЕФОНИИ НА ПРОИЗВОДСТВЕ

Гурин И.А.

*ФГАОУ ВПО «Уральский федеральный университет  
имени первого Президента России Б.Н. Ельцина»,  
г. Екатеринбург, Россия*

При современном уровне развития технологий и мобильности нашего общества человек каждый день прибегает к услугам связи. Основные средства связи в данный момент – телефония и Интернет. Телефония считается более дорогим средством связи и обходится пользователю, как правило, дороже. Особенно с ценой на услуги мы сталкиваемся, когда используем междугородные и международные направления, либо когда наша работа связана с об-

щением. Целью работы является изучение процесса автоматизации телефонных вызовов. В качестве итога данной работы выступает сервис голосовых открыток, позволяющий выбрать и отправить на мобильный телефон голосовое поздравление или розыгрыш. Адрес сервиса <http://wapring.ru>, дизайн сервиса оптимизирован для просмотра на мобильных браузерах.

### ***Основы процесса автоматизации***

Для автоматизации телефонии на производстве используются автоматические телефонные станции (АТС), позволяющие выстраивать маршруты на входящие и исходящие вызовы. АТС могут быть представлены в виде аппаратных или программных решений. Мы можем использовать мощные и дорогие решения, например, от компании Cisco, либо настроить программируемый модуль сами.

Выпускаемые в настоящее время аппаратные цифровые АТС и мини-АТС Panasonic, Samsung, LG и другие имеют практически одинаковый набор функций, включающий возможности наращивания внешней и внутренней емкости за счет подключения дополнительного оборудования, а также подключения различных типов телефонов (аналоговых, цифровых, DECT, IP) и протоколов связи.

Программные АТС – многочисленные бесплатные или коммерческие системы. Среди них всех особое место занимает Asterisk. Asterisk – свободное решение компьютерной телефонии с открытым исходным кодом от компании Digium, первоначально разработанное Марком Спенсером. Приложение работает на операционных системах Linux, FreeBSD, OpenBSD и Solaris [1]. Основные возможности и преимущества Asterisk [2]:

- Переадресация. В системе Asterisk можно сделать переадресацию по времени суток, присутствия на рабочем месте, переадресовать одновременно на несколько телефонов, составить индивидуальный алгоритм переадресации.
- Нет надобности в выделенной линии для факса. Прием факсов обеспечивается на всех номерах с автоматической отсылкой на e-mail.
- Голосовая почта. Количество голосовых сообщений не лимитировано и зависит от емкости жесткого диска. Помимо проверки голосовой почты по телефону через систему DTMF в Asterisk автоматически работает отправка голосовых сообщений на e-mail, а также существует web-интерфейс для управления своей почтой, что неоспоримо в случае переполнения голосового почтового ящика.
- Единая сетевая инфраструктура и сокращение техперсонала. В отличие от традиционных систем связи, требующих прокладку отдельной кабельной сети для телефонии, IP-PBX Asterisk работает по протоколу TCP/IP поверх существующих сетей данных. Административной телефонной инфраструктуры занимается системный или сетевой администратор, а не связист, так как IP-PBX – это сетевое приложение, не требующее особых знаний в телефонии.
- VoIP (Voice over IP). VoIP – голосовая связь по протоколу IP. Под IP-телефонией подразумевается набор коммуникационных протоколов, технологий и методов, обеспечивающих двустороннее голосовое общение (в том числе при видеообщении) по сети Интернет или по любым другим IP-сетям. Сигнал по каналу связи передается в цифровом виде и, как правило, перед передачей преобразовывается (сжимается) с тем, чтобы удалить избыток информации [3].
- Другие функциональные возможности, такие как запись разговоров, постановка звонков в очередь, индивидуальная и корпоративная записная книжка прямо в станции и служба каталога, предоплаченные карточки, интеграция с базами данных, функция «обратный звонок», а также открытый интерфейс разработчика API, позволяют максимально тесно интегрировать телефонию во все коммуникационные каналы компании.

### ***Процесс установки***

Таким образом, остановимся на использовании системы Asterisk. На первом этапе были произведены ознакомление и установка ПО. Установка производилась в операционной системе Centos 6.2. Список проблем, с которыми можно столкнуться при установке:

- невозможность конфигурирования модулей при установке из репозитория. Таким образом, требуется компиляция и установка их исходных кодов;
- большое количество зависимостей требует предварительной настройки операционной системы.

Далее представлен список команд, выполненных при установке.

```
yum install gcc-c++
yum install libtermcap-devel
yum install libxml2-devel
yum install sqlite-devel
```

```
make menuconfig
make
make install
make samples
make config.
```

Результатом работы будет являться установленный, но ненастроенный Asterisk. На этом процесс установки не заканчивается. Дополнительно для воспроизведения голосовых потоков в формате MP3 требуется установка mpg123. Mpg123 в свою очередь требует наличия libtool-ltdl-1.5. Данные пакеты можно найти во многих популярных репозиториях.

### **Процесс настройки Asterisk**

Второй этап заключается в конфигурировании Asterisk. В свою очередь конфигурирование делится на настройку доступа к VoIP-шлюзу и настройку вызова до абонента. Asterisk включает около 100 конфигурационных файлов, что создает большие затруднения для новичка. Для настройки IP-телефонии используется файл sip.conf. Далее приведена конфигурация файла настроенного шлюза voip.qip.ru.

```
[general]
srvlookup=yes
nat=yes
language=ru
register=sportsoft:password:"sportsoft@qip.ru"@qip.ru
```

```
[qip]
type=peer
username=sportsoft@qip.ru
fromuser=sportsoft
secret=password
host=qip.ru
fromdomain=qip.ru
;dtmfmode=rfc2833
context=qip
disallow=all
allow=alaw
insecure=port,invite
nat=yes
;outboundproxy=sip.qip.ru.
```

Приступим к настройке шаблона совершаемых вызовов. Настройка осуществляется в файле extensions.conf. В нем определяется обработка и маршрутизация входящих и исходящих вызовов. Этот файл управляет поведением всех соединений проходящих через АТС.

```
[call-out]
exten => 17,1,Answer
exten => 17,n,Set(CDR(userfield)=${userfield})
exten => 17,n,Set(CHANNEL(language)=ru)
```

```

exten => 17,n,MP3Player(${filename})
exten => 17,n,Wait(1)
exten => 17,n,Hangup()
;exten => 17,n,SayNumber(12).

```

Каждая секция в файле extensions.conf начинается со строки с именем секции, заключенного в квадратные скобки. Номер секции потребуется при запросе совершения вызова.

В синтаксисе файла extensions.conf каждый исполняемый шаг описывается в данном формате:

```

exten => extension,priority,Command(parameters)

```

Extension (расширение) определяется как набор команд для исполнения. Команды выполняются в порядке, который определяется их параметром приоритета. Когда происходит вызов на определенное расширение, будет выполнена команда с приоритетом 1, затем с приоритетом 2 и так далее. Использование символов, отличных от цифр, в поле приоритета, требуется при использовании приоритетов типа n (next).

Выполнение команд производится до того момента, пока не возникнет одна из следующих ситуаций [4]:

- вызывающий абонент повесил трубку;
- команда вернула код возврата -1 (индикатор ошибки);
- команды со следующим приоритетом не существует (Asterisk не будет перескакивать через недостающие приоритеты);
- вызов был переброшен на другое расширение.

Совершать исходящие вызовы в Asterisk можно 2 способами:

- Asterisk auto-dial out с использованием call-файлов. Принцип основан на том, что собранные файлы помещаются в специальную директорию, которую сканирует Asterisk. При обнаружении программой нового файл, он выполняется;
- Asterisk Manager API с использованием команды Originate. AGI (Asterisk Gateway Interface) – это встроенный в Asterisk метод выполнения внешних скриптов, который может расширить функциональность Asterisk при помощи других языков программирования. Список языков программирования: Perl, PHP, C, Pascal, Bourne Shell.

Второй способ является более гибким и надежным, но более сложным в настройке. С целью более надежной работы программы, используется второй вариант.

Работа AGI начинается с передачи списка переменных, связанных с расширением. После этого, это приложение отправляет серверу Asterisk специальные команды AGI интерфейса через стандартный поток вывода и получает результаты выполнения этих команд, через поток стандартного ввода.

Рассмотрим пример AGI-скрипта на языке программирования PHP.

```

require(ROOTPATH.'/classes/asterisk/AsteriskManager.php');

```

```

$sast = new Net_AsteriskManager(array('server' => '127.0.0.1', 'port' => '5038', 'auto_connect' => true));

```

```

$sast->login('admin', 'passwd');

```

```

$command = (
    "Action: Originate\r\n".
    "Channel: SIP/quip/+79051234567\r\n".
    "Context: call-out\r\n".
    "Exten: 17\r\n".
    "Priority: 1\r\n".
    "Timeout: 40000\r\n".
    "Async: true\r\n".
    "Callerid: +79530000018\r\n".

```

```
"Variable: userfield=1,filename=/home/sportsoft/wapring.ru/mp3/1.mp3\r\n"  
);  
$ast->Call($command);
```

Переменная `command` содержит параметры, требуемые для совершения звонка. Таким образом, настройка исходящих вызовов настроена и готова к применению.

### **Заключение**

В данной статье рассмотрена работа АТС Asterisk и одно из ее направлений работы – автоматизация исходящих вызовов. Автоматизация применяется в многочисленных системах автодозвона и оповещений. Кроме функций обычной АТС, Asterisk имеет возможность оптимизации затрат на телефонию. Снижения затрат можно достичь за счет использования самых выгодных тарифов по конкретным направлениям. Например, звонки на номера оператора МТС осуществлять с сим-карт этого же оператора, а для звонков на городские номера использовать проводную городскую телефонную линию. В обоих случаях цена за звонок может быть снижена до нуля (за исключением абонентской платы). Но это уже тема других работ.

### **Список использованных источников**

1. Википедия, свободная энциклопедия [Электронный ресурс]. Asterisk. Режим доступа: <http://ru.wikipedia.org/wiki/Asterisk>.
2. Asterisk [Электронный ресурс]. Файл `extensions.conf` – ваш план набора. Режим доступа: <http://asterisk.ru/knowledgebase/Asterisk+config+extensions.conf>.
3. Википедия, свободная энциклопедия [Электронный ресурс]. IP-телефония. Режим доступа: <http://ru.wikipedia.org/wiki/IP-телефония>.
4. Itslink.ru: IP-телефония [Электронный ресурс]. Сравнение аппаратных и программных АТС. Режим доступа: <http://itslink.ru/stati/sravnenie-apparatnykh-cifrovyykh-ats-i-programmnoi-ats-asterisk/>.

## **РАЗРАБОТКА И ВНЕДРЕНИЕ SCADA-СИСТЕМЫ В РАМКАХ ПРОЕКТА ПО СОЗДАНИЮ АСУТП И МОДЕРНИЗАЦИИ ПРОИЗВОДСТВА ХРОМОВОГО АНГИДРИДА НА ЗАВОДЕ ЗАО "РУССКИЙ ХРОМ 1915"**

**Девярых Е.А., Берковская Д.В.**

*ФГАОУ ВПО «Уральский федеральный университет  
имени первого Президента России Б.Н. Ельцина»,  
г. Екатеринбург, Россия*

В рамках технического перевооружения производства хромового ангидрида на ЗАО «Русский хром 1915» (г. Первоуральск) компания «Феррокс» внедряет автоматизированную систему управления технологическим процессом.

Техническое перевооружение заключается в переводе существующего производства хромового ангидрида периодическим способом (производительность 19 000 т/год) в производство хромового ангидрида непрерывным способом (производительность 6000 т/год) и введением раствора гипохлорита натрия в бисульфитную травку монокроматных растворов.

Целями разработки SCADA системы, как части АСУТП являются:

- улучшить технико-экономические показатели производства;
- автоматизация процесса утилизации отходов производства хромового ангидрида в соответствии с современными экологическими нормами;
- повышение выпуска годного по качеству продукта;
- контроль материальных потоков;
- улучшение условий труда технологического и обслуживающего персонала.